

標準単体上の最小 2 乗問題に対する対数正則化と近接分離法

田中未来 数理・推論研究系 計算推論グループ, 統計的機械学習研究センター 助教

1 考える問題

$$\begin{aligned} & \text{最小化} \quad \frac{1}{2} \|\mathbf{Ax} - \mathbf{b}\|_2^2 + r(\mathbf{x}) \\ & \text{制約条件} \quad \mathbf{1}^\top \mathbf{x} = 1, \mathbf{x} \geq \mathbf{0} \end{aligned} \quad (1)$$

- $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{b} \in \mathbb{R}^m$: 入力データ
- $\mathbf{x} \in \mathbb{R}^n$: (最適化問題の意味での) 変数
- $r: \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$: 正則化関数

2 材料科学分野での応用 (Tanaka et al., '17)

反応速度の異なる複数の物質からなる混合物の光化学反応

- b_i : 時刻 t_i における混合物の吸光度の測定値
- a_{ij} : 時刻 t_i における成分 j の吸光度の理論値
- x_j : 成分 j の混合比率

3 正則化の比較

- ホワイトノイズを仮定, i.e., $\{b_i - \mathbf{a}_i^\top \mathbf{x}\}_{i=1}^m \stackrel{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)$
 - 問題 (1) と等価な MAP 推定問題を考える
- ⇒ L_2 正則化や L_1 正則化は制約条件と不整合

表 1: L_2 , L_1 , 対数正則化と対応する事前分布

正則化項 $r(\mathbf{x})$	対応する事前分布
0 (正則化なし)	$\mathbf{x} \sim$ (標準単体上の一様分布)
$(\gamma/2) \ \mathbf{x}\ _2^2$ (L_2 正則化)	$\{x_j\}_{j=1}^n \stackrel{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2/\gamma)$
$\gamma \ \mathbf{x}\ _1$ (L_1 正則化)	$\{x_j\}_{j=1}^n \stackrel{i.i.d.}{\sim} \text{Laplace}(0, \sigma^2/\gamma)$
$-\sum_{j=1}^n \gamma_j \log x_j$ (対数正則化)	$\mathbf{x} \sim \text{Dir}(\gamma/\sigma^2 + \mathbf{1})$

4 アルゴリズム

次の対数正則化問題を効率よく解くアルゴリズムを考える

$$\begin{aligned} & \text{最小化} \quad \frac{1}{2} \|\mathbf{Ax} - \mathbf{b}\|_2^2 - \sum_{j=1}^n \gamma_j \log x_j \\ & \text{制約条件} \quad \mathbf{1}^\top \mathbf{x} = 1, \mathbf{x} > \mathbf{0} \end{aligned} \quad (2)$$

4.1 加速近接勾配法 (APGM)

$$f(\mathbf{x}) = \frac{1}{2} \|\mathbf{Ax} - \mathbf{b}\|_2^2, \quad g(\mathbf{x}) = -\sum_{j=1}^n \gamma_j \log x_j + \iota \left(\begin{array}{l} \mathbf{1}^\top \mathbf{x} = 1 \\ \mathbf{x} > \mathbf{0} \end{array} \right)$$

とすると問題 (2) は $f(\mathbf{x}) + g(\mathbf{x})$ の無制約最小化と等価

—— 加速近接勾配法 (APGM) ——

for $k = 1, 2, \dots$ (収束するまで)
 $\mathbf{y}^{(k+1)} = \mathbf{x}^{(k)} + \omega^{(k)}(\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)})$
 $\mathbf{x}^{(k+1)} = \text{prox}_{\lambda^{(k)}g}(\mathbf{y}^{(k+1)} - \lambda^{(k)} \nabla f(\mathbf{y}^{(k+1)}))$

$\text{prox}_{\lambda g}(\mathbf{v}) = \arg\min_{\mathbf{x}} \{g(\mathbf{x}) + (1/2\lambda) \|\mathbf{x} - \mathbf{v}\|_2^2\}$ は 2 分法で

4.2 交互方向乗数法 (ADMM)

$$\begin{aligned} h(\mathbf{x}) &= \frac{1}{2} \|\mathbf{Ax} - \mathbf{b}\|_2^2 + \iota(\mathbf{1}^\top \mathbf{x} = 1) \\ l(\mathbf{x}) &= -\sum_{j=1}^n \gamma_j \log x_j + \iota(\mathbf{x} > \mathbf{0}) \end{aligned}$$

とすると問題 (2) は $h(\mathbf{x}) + l(\mathbf{x})$ の無制約最小化と等価

—— 交互方向乗数法 (ADMM) ——

for $k = 0, 1, 2, \dots$ (収束するまで)
 $\mathbf{x}^{(k+1)} = \text{prox}_{\lambda^{(k)}h}(\mathbf{z}^{(k)} - \mathbf{u}^{(k)})$
 $\mathbf{z}^{(k+1)} = \text{prox}_{\lambda^{(k)}l}(\mathbf{x}^{(k+1)} + \mathbf{u}^{(k)})$
 $\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \mathbf{x}^{(k+1)} - \mathbf{z}^{(k+1)}$

- $\text{prox}_{\lambda l}(\mathbf{v})$ の計算は容易 (よく知られている)
- $\text{prox}_{\lambda h}(\mathbf{v})$ は閉じた解を持つが $(\mathbf{A}^\top \mathbf{A} + (1/\lambda)\mathbf{I})^{-1}$ が必要

4.3 線形化交互方向乗数法 (LADMM)

交互方向乗数法と違って $\mathbf{x}^{(k+1)}$ を次の問題の最適解とする

$$\begin{aligned} & \text{最小化} \quad h(\mathbf{x}) + \frac{1}{2\lambda^{(k)}} \|\mathbf{x} - \mathbf{z}^{(k)} + \mathbf{u}^{(k)}\|_2^2 + \frac{1}{2} \|\mathbf{x} - \mathbf{x}^{(k)}\|_G^2 \\ & \text{制約条件} \quad \mathbf{1}^\top \mathbf{x} = 1 \end{aligned}$$

$\mathbf{G} = \sigma_{\max}(\mathbf{A})^2 \mathbf{I} - \mathbf{A}^\top \mathbf{A}$ とすると最適解の計算は容易

5 計算機実験

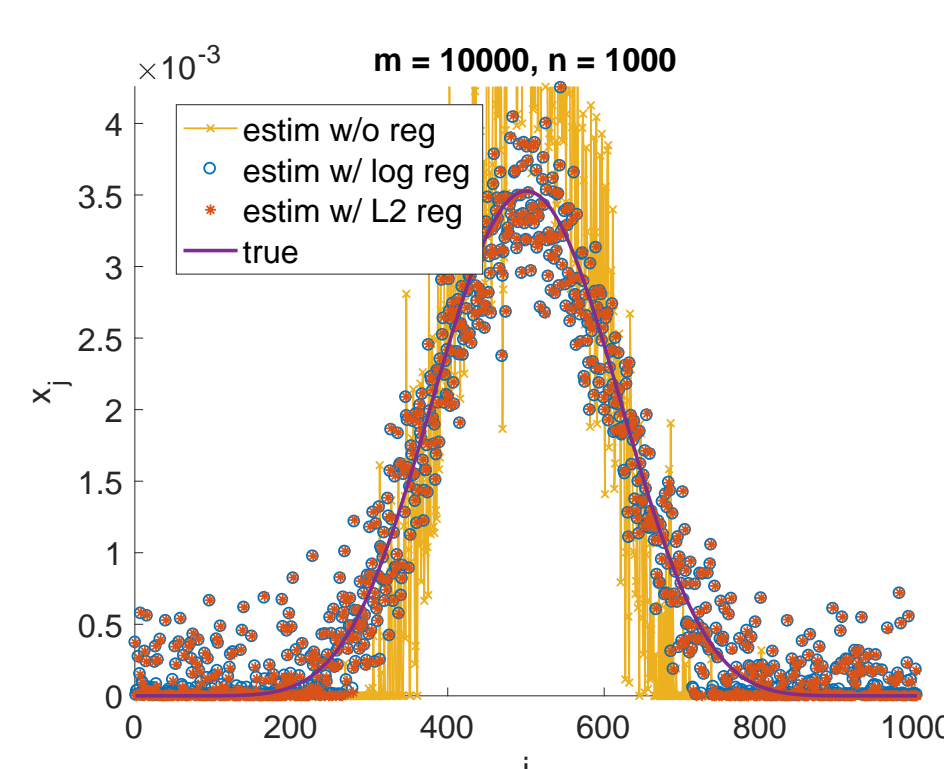
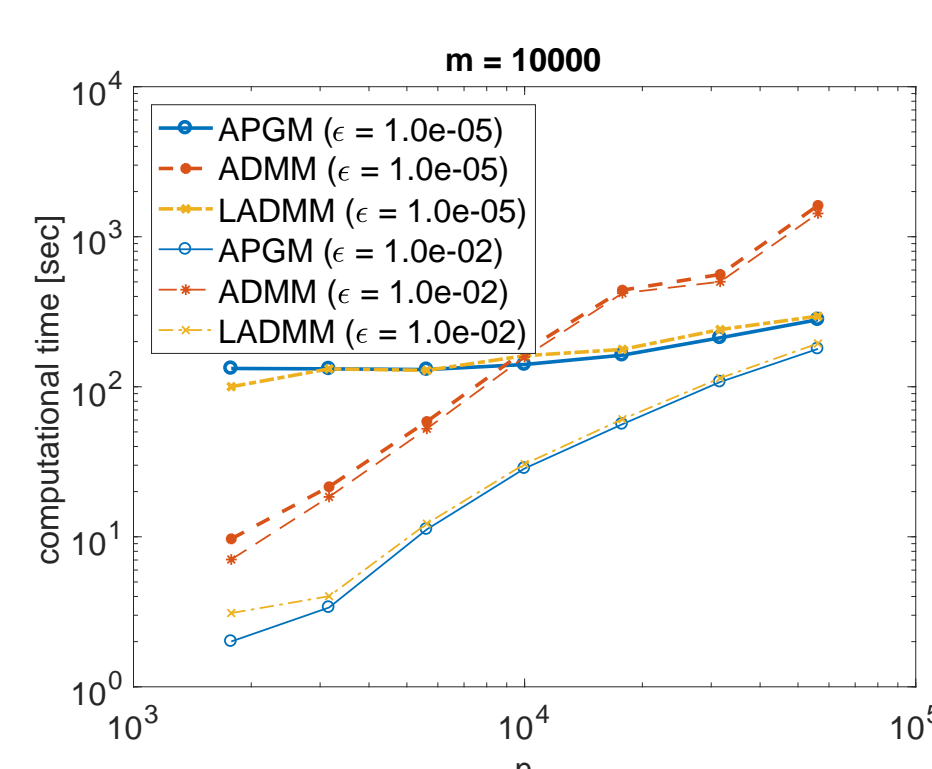


図 1: 問題例の規模と終了条件がアルゴリズムの計算時間に与える影響

図 2: 真の値 \mathbf{x}^* と推定値 $\hat{\mathbf{x}}$

- APGM, LADMM は高速だが最適解付近での収束が遅い
- ADMM は \mathbf{A} の特異値分解に時間がかかるが収束は速い

謝辞

本研究は武田朗子氏との共同研究です。廣瀬雅代氏の助言も参考にしました。

参考文献

M. Tanaka et al.: *AIP Advances* **7** (2017) 015041.

